



# Handling Topological Changes during Elastic Registration: Application to Augmented Reality in Laparoscopic Surgery

Christoph Joachim Paulus, Nazim Haouchine, Seong-Ho Kong, Renato Vianna Soares, David Cazier, Stéphane Cotin

## ► To cite this version:

Christoph Joachim Paulus, Nazim Haouchine, Seong-Ho Kong, Renato Vianna Soares, David Cazier, et al.. Handling Topological Changes during Elastic Registration: Application to Augmented Reality in Laparoscopic Surgery. International Journal of Computer Assisted Radiology and Surgery, 2016, 12, pp.461-470. 10.1007/s11548-016-1502-4 . hal-01397409

**HAL Id: hal-01397409**

**<https://inria.hal.science/hal-01397409>**

Submitted on 15 Nov 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Handling Topological Changes during Elastic Registration

## Application to Augmented Reality in Laparoscopic Surgery

Christoph J. Paulus · Nazim Haouchine · Seong-Ho Kong · Renato Vianna Soares ·  
David Cazier · Stephane Cotin

Received: November 11, 2016/ Accepted: date

**Abstract** *Purpose:* Locating the internal structures of an organ is a critical aspect of many surgical procedures. Minimally invasive surgery, associated with augmented reality techniques, offers the potential to visualize inner structures, allowing for improved analysis, depth perception or for supporting planning and decision systems.

*Methods:* Most of the current methods dealing with rigid or non-rigid augmented reality make the assumption that the topology of the organ is not modified. As surgery relies essentially on cutting and dissection of anatomical structures, such methods are limited to the early stages of the surgery.

We solve this shortcoming with the introduction of a method for physics-based elastic registration using a single view from a monocular camera. Singularities caused by topological changes are detected and propagated to the pre-operative model. This significantly improves the coherence between the actual laparoscopic view and the model, and provides added value in terms of navigation and decision-making, e.g. by overlaying the internal structures of an organ on the laparoscopic view.

*Results:* Our real time augmentation method is assessed on several scenarios, using synthetic objects and real organs. In all cases, the impact of our approach is demonstrated, both qualitatively and quantitatively<sup>1</sup>.

*Conclusion:* The presented approach tackles the challenge of localizing internal structures throughout a complete

surgical procedure, even after surgical cuts. This information is crucial for surgeons to improve the outcome for their surgical procedure and avoid complications.

### 1 Introduction

The last decades show considerable advances in the introduction of augmented reality during surgery<sup>15</sup>. In particular, the scientific community and clinicians have been focusing on minimally invasive surgery (MIS). This type of surgery has gained popularity and became a well-established procedure thanks to its benefits for the patient in term of infection risks reduction and shortened recovery time. However, it remains complex from a surgical point of view, mainly because of the reduced field of view that significantly impacts depth perception and surgical navigation.

These limitations are reduced with computer guidance by overlaying a three-dimensional pre-operative model of the patient's anatomy onto the images provided by laparoscopic cameras. This augmentation typically involves two steps: the visualization of the anatomical or pathological structures present in the medical images, and the registration of the pre-operative model onto the intra-operative view.

Common registration processes are either interactive or automatic. An interactive registration requires the manipulation of the virtual organ until its projection on the laparoscopic images matches the current shape of the organ. This task is time consuming and the approach is limited to rigid registration. An interactive registration requires the selection of landmarks on both the virtual pre-operative model and the intra-operative view whose quality and pairwise matching can hardly be guaranteed.

For these reasons, many works now focus on automatic non-rigid registration<sup>28;23;9</sup>. Till now, except for Petit *et al*<sup>21</sup> and our previous work<sup>18;17</sup> none of the proposed methods

---

Christoph Paulus, Nazim Haouchine and Stephane Cotin  
Inria

Renato Vianna Soares  
IHU Strasbourg

Seong-Ho Kong  
IHU Strasbourg and Seoul National University Hospital

David Cazier  
University of Strasbourg - ICube lab CNRS

<sup>1</sup> <http://www.open-cas.org/?q=PaulusIJCARS16>

consider cuts, and therefore topological changes, during the registration process. We advocate that the ability to detect surgical cuts in the laparoscopic images and to apply them to the pre-operative model in real time, is a key feature for augmented reality applications. The present paper addresses the real-time elastic registration of 3d-models subject to topological changes, observed from a single view by a monocular camera, extending the seminal work of Paulus *et al*<sup>18</sup>. To our knowledge, this is the first work that proposes an automatic update of the topology and placement of the internal structures of the organs in response to topological changes detected in the registered models.

## 2 Related works

### 2.1 Augmentation of deformable objects

In the context of non-rigid objects, two main behaviors can be identified: inextensible ones such as clothes and elastic ones such as soft tissues. First, inextensible surfaces have been considered with the exploitation of distance constraints. The surface deformation is essentially computed through parametric geometrical models<sup>2;22</sup>. Approaches based on learning methods have also been considered. The solution is estimated from a representative sample of possible shapes using a dimensionality reduction process<sup>25;24</sup>.

As regards elastic behaviors, geometrical properties cannot be exploited. The closed-form solution constrained by shading information<sup>14</sup> can capture stretching surfaces and yields good results. But the method assumes a Lambertian surface with a single point light source what greatly limits the possible applications.

More recently, Agudo *et al*<sup>1</sup> describe elastic shapes with physical models and combine the finite element method with an extended Kalman filter. Other approaches try to minimize a stretching energy<sup>13</sup> that unifies geometric and mechanical constraints using a projective camera and locally linear deformations. In a similar way, non-linear elastic models<sup>7</sup> have been used to augment highly elastic objects. The model is constrained by features extracted from a single view camera and a set of boundary points.

The aforementioned methods cope with the registration or augmentation of surfaces. Augmenting 3D objects requires a more advanced acquisition process. For instance, a stereoscopic visual tracking<sup>8</sup> can be associated with a physical model to augment a volumetric object. A regularization mechanism makes the method more robust to errors in the tracking process. In a similar way, Leiza *et al*<sup>12</sup> exploit a depth camera to capture a 3D deformed shape.

The approach presented by Petit *et al*<sup>20</sup> fits the point cloud provided by an RGB-D sensor with a tetrahedral mesh using a geometrical point-to-point coupling. Linear elastic external forces are applied on an adapted corotational finite

element model increasing the weight of the forces applied to the contour of the object. The method supports occlusion of the object and strong or highly elastic deformations. It has been extended, but only for non-elastic objects, to cope with fractures<sup>21</sup> that appear when the maximal eigenvalue of the stress tensor exceeds the toughness of the material.

In the context of Computer Assisted Surgery, patient-specific biomechanical models demonstrated their relevance for volume registration. They take into account anisotropic elastic deformation to infer in-depth structure motion<sup>28;9</sup>. Pratt *et al* use a 4D scan<sup>23</sup> of the heart and a biomechanical model to couple the surface motion with external forces that emanate from camera data.

Nevertheless, cuts and resection are essential in surgical procedures. The virtual model on which the augmented view relies has to be updated to allow a sound localization of the internal structures or tumors. In the context of image-guided neurosurgery, Ferrant *et al*.<sup>6</sup> handle the registration issues induced by tumor resection with the removal of the elements of the brain model that contains the resected tumor.

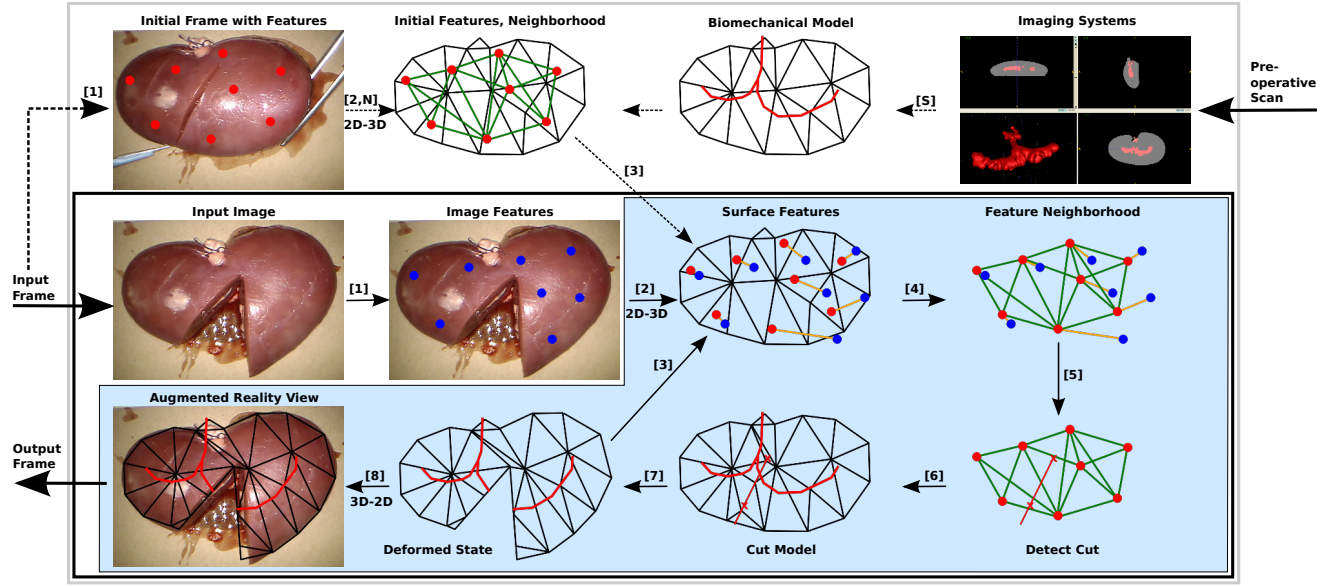
### 2.2 Simulation of cuts, fractures and tears

The simulation of surgical procedures raises specific issues and challenges. The organs undergo elastic – and sometimes plastic – deformations and, beyond that, are subject to cuts, tears or cauterization. The used biomechanical models must therefore support topological changes or, in other words, updates in the connectivity of the underlying meshes. The simulation of cutting, fracture and tearing remains an active research domain in computer graphics. A good overview is provided by Wu *et al*<sup>29</sup>. In the following, we focus on the Finite Element Method that is used in this paper.

The composite finite element method<sup>30</sup> (CFEM) uses a fine grid for the visualization and a coarse one for the deformations. The cuts are performed on the fine grid and propagated to the coarse mesh. This approach allows real-time performance but prevent partial cuts of the coarse elements to be visible.

The snapping method<sup>16</sup> consists in moving the vertices of the original mesh toward the separation surface before disconnecting the object along these *snapped* vertices. The cuts are limited to the topology of the original object that is often coarse to ensure real-time performances.

Local remeshing overcome this limitation and allow for partial cuts of elements<sup>11</sup>. They may be combined with snapping to avoid instabilities in the simulation and to limit the increase degrees of freedom, as proposed by Paulus *et al*<sup>19</sup>.



**Fig. 1** Pipeline of our method: [S] Segmentation, [1] identify image features  $p_{l,f}$  (see 4.2), [2] map image features on model (see 4.2, 5.2), [N] construct feature neighbourhood  $C_F^0$  (see 5.2), [3] use initial/updated model  $(P_V, C_V)$  and compare image  $p_{l,f}$  and surface features  $p_{l,F}$  by [4] calculating the measure  $\mu_{lm}$  on the neighborhood  $C_F$ , [5] detect outliers, insert cut points  $p_C$ , expand cut points to cut lines, [6] expand cut line to cut surface  $S$  (see 5.2), update topology of model and internal structures (see 5.3, 5.4), [7] solve minimization problem (5) of 4.3, [8] display model on output frame; the dashed flashes are performed at initialization; one frame with image features can be used several times, repeating the steps [2]-[8] in the blue area.

### 3 Method overview

In the following, we use  $X_E^S$  to name a variable  $X$  of the entity  $E$  in the simulation state  $S$ . The variable  $X$  can denote either one point  $p$ , a set of points  $P$ , the connectivity  $c$  of one FEM element or the connectivity  $C$  of a set of elements. The name  $E$  denotes the entity the variable belongs to. In the case of a 3D object we use  $V$  for the volume,  $\partial V$  for the surface of the volume,  $I$  for the internal structures and  $F$  for the surface features. On the other hand, in a 2D image, we use  $f$  for the features.

Finally, the status  $S$  can be either the initial state 0, the current state  $t$  or the target state 1. Let us point out that the *targets* denote here the features extracted from the images in the video stream that the virtual model tries to follow. Thus the targets change with each video frame. In addition, the current state is related to the actual position of the virtual model while it moves towards the target.

Before a surgical intervention, the organ and its internal structures are segmented from the pre-operative images. Separate meshes are built for each of the considered 3D objects. The virtual organ is discretized in a set of tetrahedra  $C_V^0$  whose elements are denoted  $c_V^0$ . Those elements connect the vertices  $P_V^0$  of the volume mesh that models the organ. The internal structures are discretized as a surface mesh  $C_I^0$  that connects the vertices  $P_I^0$ . To geometrically bind the two initial sets of points, the  $P_I^0$  are expressed as barycentric coordinates of the  $(P_V^0, C_V^0)$ . Finally, constitutive laws and a set

of parameters are chosen to approximate the elastic behavior of the organ and internal structures.

Features points of the real organ are identified in the laparoscopic view and are registered to the virtual model in its initial position. We denote this set  $P_F^0 = \{p_{l,F}^0\}$ . During the surgical intervention, the detected features are tracked in the video stream. They form the target points  $P_F^1 = \{p_{l,F}^1\}$ . The real and virtual organs are coupled, by means of these two sets of features points: the tracked features  $P_F^1$  and their initial registration  $P_F^0$  that move according to the deformation of the virtual model and whose current positions are  $P_F^t$ .

Our method captures the deformations, and detects cuts and tears, through an analysis of the displacement field of these two point clouds. Detected cuts are reproduced on the virtual organ and the internal structures are updated providing additional information for the surgeon during the advancement of the surgical procedure.

### 4 Elastic registration

#### 4.1 Physically-based model

The choice of the constitutive law determines the set of deformations that can be represented while discriminating non-plausible configurations that could be induced by wrong target surface feature point cloud  $P_F^1$ .

The constitutive law of St. Venant-Kirchhoff represents hyperelastic material behaviour and is often used to simulate non-linear deformations in real-time. However, beyond other unintended side effects, due to the monotonicity in compression, it can break down under extreme compression<sup>27</sup>. Thus we apply the corotational linear elasticity that secures the rotational invariance and thus non-linear characteristics, while keeping the simplicity of the stress-deformation relationship in linear materials<sup>26</sup>. The corotational finite element method uses the polar decomposition of the deformation gradient  $\mathbf{F} = \mathbf{R}\mathbf{U}$  to construct a new strain measure

$$\boldsymbol{\varepsilon}_c = \mathbf{U} - \mathbf{I} \quad (1)$$

The strain global energy  $W_c$  estimated for each element  $c$  in the FE model is given by the equation:

$$W_c = \frac{\lambda}{2} [\text{tr}(\boldsymbol{\varepsilon}_c)]^2 + \mu \text{tr}(\boldsymbol{\varepsilon}_c^2) \quad (2)$$

where  $\lambda$  and  $\mu$  are Lamé coefficients and can be computed thanks to the elastic parameters of the material  $E$  and  $\nu$ .  $E$  is the Young's modulus and is a measure of the stiffness of the material while  $\nu$  is the Poisson's ratio which estimates the compressibility of the material.

We obtain the global internal elastic energy by accumulating the strain energies  $W_c$  of the elements:

$$W_I(P_V, C_V^0) = \sum_{c \in C_V^0} W_c \quad (3)$$

## 4.2 Identification and tracking of features

An initial registration of the biomechanical model to the video stream can either be performed manually, using contour based approaches<sup>5</sup> or using anatomical landmarks.

From the first frame of a monocular video stream in a single view position we extract 2D features in their initial positions  $p_{l,f}^0 \in P_f^0$  using the Speeded-up Feature Detector. These 2D positions are mapped to the surface of the initially registered model and the surface feature points  $p_{l,f}^0 \in P_f^0$  are expressed as barycentric coordinates of  $(P_V^0, C_V^0)$ . For the frame-to-frame tracking the Lucas-Kanade optical flow yields 2D features  $p_{l,f}^1 \in P_f^1$  and with the mapping to the surface, the target surface feature points  $p_{l,f}^1 \in P_f^1$ .

## 4.3 Energy minimization problem

In the simulation, the *current* features points  $p_F^t$  are mapped to the FEM mesh and move according to the mesh nodes  $P_V^t$ , so we can write  $p_{l,f}^t = p_{l,f}^t(P_V^t) \in P_F^t(P_V^t)$ .

The coupling of the real and virtual models is obtained with the introduction of spring forces between each surface

feature point  $p_{l,f}^t(P_V^t)$  and target surface feature point  $p_{l,f}^1$ , which accumulate to the stretching energy:

$$W_S(P_F^t(P_V^t), P_F^1) = \sum_l \frac{1}{2} k_l \|p_{l,f}^t(P_V^t) - p_{l,f}^1\|^2 \quad (4)$$

The parameters  $k_l$  are experimentally chosen and are in the same order of magnitude as the Young's modulus of the deformable object. The updated set of vertices  $P_V^t$  is obtained by solving the minimization problem between internal elastic energy and stretching energy:

$$\underset{P_V^t}{\text{argmin}} (W_I(P_V^t, C_V^0) + W_S(P_F^t(P_V^t), P_F^1)) \quad (5)$$

The surface feature points  $P_F^t(P_V^t)$  and the internal structures  $P_I^t(P_V^t)$  are updated applying their initial barycentric coordinates to the new positions of the  $P_V^t$ .

Since we use a dynamic simulation, there is no need for boundary conditions to obtain a stable problem. The minimization problem is solved every time step using the conjugate gradient method and we apply the Euler implicit method for the time integration.

## 5 Topological changes

This section presents the core of the proposed method. We first give a mathematical formulation of the problem, stating our assumptions and our goal: provide an updated virtual organ, including its internal structures, even when dealing with large deformations and/or topological changes. Then, we describe the detection of a cut, the way it is processed and the update of the internal structures.

### 5.1 Problem formulation

We assume that the initial positions of the surface feature points  $P_F^0$ , the virtual organ  $P_V^0$ , the internal structures  $P_I^0$  and the target positions of the surface feature points  $P_F^1$  are given. The tetrahedral elements  $C_V^0$  – respectively the triangular ones  $C_I^0$  – connect  $P_V^0$  (and respectively  $P_I^0$ ) to the virtual organ (and the internal structures).

Our objective is to update the shape (geometry and topology) of the virtual organ  $(P_V^0, C_V^0)$ , based on the final position of the target surface feature points  $P_F^1$ . This includes the deformation and update of the topology of the model  $(P_V^t, C_V^t)$  and the internal structures  $(P_I^t, C_I^t)$ . With the previous notation the problem can be formulated as:

$$\text{Given } P_F^0, (P_V^0, C_V^0), (P_I^0, C_I^0) \text{ and } P_F^1 \quad (6)$$

$$\text{Find } (P_V^t, C_V^t) \text{ solving (5)} \quad (7)$$

$$\text{and minimizing } \|P_F^t(P_V^t) - P_F^1\|_F \quad (8)$$

When a ground truth of the surface of the volumetric model ( $P_{\partial V}^1, C_{\partial V}^1$ ) or of the internal structures ( $P_I^1, C_I^1$ ) are given, the problem could be complemented by:

$$\text{Minimize } \|(P_{\partial V}^t(P_V^t), C_{\partial V}^t) - (P_{\partial V}^1, C_{\partial V}^1)\|_{\partial V} \quad (9)$$

$$\text{Minimize } \|(P_I^t, C_I^t) - (P_I^1, C_I^1)\|_I \quad (10)$$

The choice of the norms depends on the application and will be discussed in the section 6. To solve the problem, the displacement field between the current features  $P_F^t(P_V^t)$  and target features  $P_F^1$  is analyzed. Discontinuities in this field reveal an inconsistency between the biomechanical model ( $P_V^t, C_V^t$ ) and the target surface feature points  $P_F^1$ . Such an inconsistency can be triggered by a cut of the real model that moves the target surface feature points  $P_F^1$ .

## 5.2 Cut detection

**Detection of a cut region:** The minimization problem (5) is solved at each time step, resulting in an update of the biomechanical model ( $P_V^t, C_V^t$ ). When a smooth deformation of the organ occurs, the biomechanical model ( $P_V^t, C_V^t$ ), and with it the surface feature points  $P_F^t(P_V^t)$ , can properly follow the target surface feature points  $P_F^1$ . As soon as a cut occurs, observations of the surface feature points are twofold: First, the uncut model ( $P_V^t, C_V^t$ ) becomes unable to adapt to the motion enforced by the stretching energy. As a result, the vector between the current and the target positions  $d_l(P_V^t, P_F^1) = p_{l,F}^t(P_V^t) - p_{l,F}^1$  diverges around the cut. Secondly, the distance between two target feature points  $p_{l,F}^1$  and  $p_{m,F}^1$  that lie on two different sides of the cut increases much more than the average distance between feature points.

To exploit these observations, we call two feature points  $p_{l,F}^0$  and  $p_{m,F}^0$  neighbors iff their initial Euclidean distance  $\delta_{lm,F}^0 = \|p_{l,F}^0 - p_{m,F}^0\|$  is smaller than a given radius  $r_{P_F^0}$ , that is related to the density of the features. The Euclidean distances  $\delta_{lm,F}^t$  and  $\delta_{lm,F}^1$  are defined respectively and will be used in the following. The neighborhood information is stored in the graph  $C_F^0 = \{(l, m) | \delta_{lm,F}^0 < r_{P_F^0}\}$  (see figure 1.[N]).

In the region of a cut, the physical model prevents the surface feature points  $P_F^t(P_V^t)$  from moving towards their targets  $P_F^1$  and the vectors  $d_l(P_V^t, P_F^1)$  and  $d_m(P_V^t, P_F^1)$  – simply denoted  $d_l$  and  $d_m$  in the following – point in two different directions. The similarity of the displacement field of neighboring features is evaluated through the Euclidean norm  $\|d_l - d_m\|$ , taking account of the first observation. For the second observation, we evaluate the ratio  $\delta_{lm,F}^1 / \delta_{lm,F}^0$  of the initial and final distance of neighboring surface feature points. An increase indicates either an elongation of the object or a separation due to a cut or a tear. These two cases are distinguished by comparing this ratio with its average in the neighborhood.

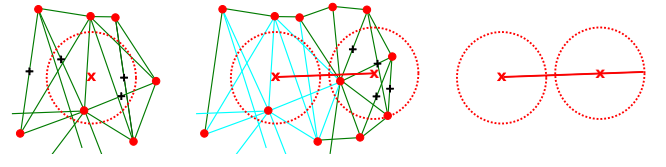
Summarizing, we define the measure

$$\mu_{lm} = \frac{\delta_{lm,F}^1}{\delta_{lm,F}^0} \|d_m - d_l\|, \forall (l, m) \in C_F^0 \quad (11)$$

and we denote  $\mu_\emptyset$  the mean value over  $C_F^0$ . With a scenario dependent threshold  $\tau$  we can identify outliers – i.e. divergent neighbor features – with  $\mu_{lm} > \tau \mu_\emptyset, (l, m) \in C_F^0$ , that lie in a region where a cut or tear is likely to occur.

**Insertion of cut points:** To detect the cuts, we first search for couples of neighboring feature points  $(l, m)$  that are outliers for the measure  $\mu_{lm}$ . For two outliers  $(l_0, m_0), (l_1, m_1)$  with  $\mu_{l_0 m_0} > \mu_{l_1 m_1} > \tau \mu_\emptyset$  the cut should be closer to the  $(l_0, m_0)$  couple. As soon as the number of outliers exceeds a given threshold  $n$ , we insert a cut point  $p_C$  at the averaged barycenter of the outliers weighted with their respective measure.

Then, to avoid the insertion of another cut point in the next time step at the same location, we delete from the set  $C_F^t$  of neighbors the couples that cross the sphere of the radius  $r_{p_C}$  around the inserted cut point (see figure 2 left, middle). After the update of the neighbors, the measure defined in (11) is only evaluated on  $C_F^t$ .



**Fig. 2** From left to right: insertion of a cut point based on the barycenters of identified outliers (black crosses); update of the connectivity information (turquoise connections are not used any more), insertion of a second cut point yields a cut line (polygon); cut polygon is extended until the end of its radius  $r_{p_C}$

With the first insertion of a cut point, we introduce a sequence of cut points  $\{p_{i_0, C}, \dots, p_{i_n, C}\}$  with  $i_0 = i_n$ , that extends to a cut polygon continuously, i.e. new cut points can be inserted before the first and after the last extremity. New cut points are inserted before or after the nearest extremity (figure 2 middle) and the polygon fits the widening of the cut.

**Expansion to cut surface:** Using the direction of the camera or a predefined vector, the separation polygon can be extruded to a separation surface  $S$ , which can be forwarded to any cutting algorithm<sup>29</sup>.



**Optimizations:** The parameters  $n$  and  $r_{pc}$  allow to balance between the precision of the cut and the robustness of the cut detection. Setting these parameters on a high value, i.e. aiming rather at the robustness than at the precision of the cut detection, reduces the ability of the detection to react instantly on an emerging cut. That means, the detection has the tendency to lag behind. To overcome this negative side effect, we insert another cut point located at the intersection between the line connecting the last and the current cut point and the sphere around the current cut point (figure 2 right).

The surface feature points do not go beyond the boundary, i.e. inserted cut points stop before the boundary of the object, preventing the occurrence of complete cuts. The preventive step of the last paragraph results in a cut over the boundary which alleviates the problem without adding a parameter.

### 5.3 Simulation of cuts

The presented method to detect topological changes is independent of the separation algorithm<sup>29</sup>. In our work we use a remeshing approach combined with a node snapping technique<sup>19</sup> which is presented below.

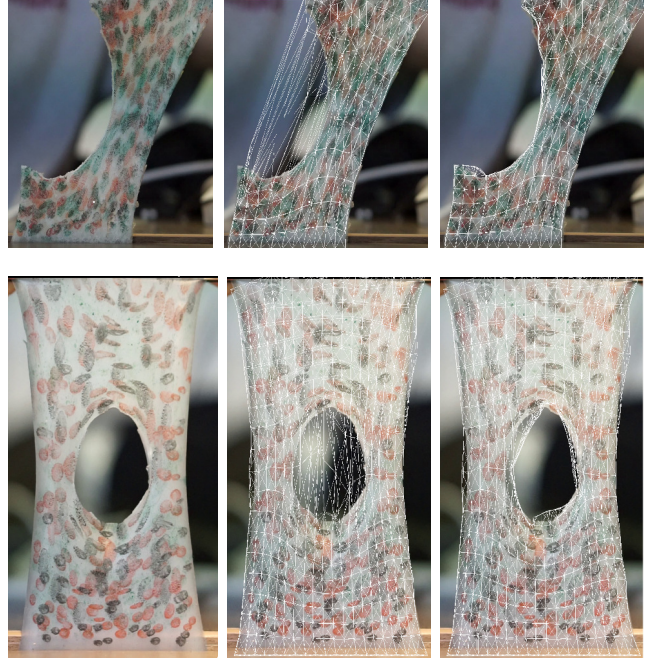
The cutting algorithm updates a FEM mesh at a predefined or detected cut (or separation) surface  $S$  in two steps. First the cut surface is sampled at the level of the edges of the FEM mesh, which allows to identify vertices that are close to the cut and need to be snapped.

After that, in other cases, a local remeshing is performed: tetrahedra adjacent to a cut edge are replaced to introduce a vertex inside every tetrahedron traversed by the separation surface. Then faces which are adjacent to the cut edge are flipped to insert new edges between the points on the separation surface. Finally, the edge that crosses the surface is deleted and triangles inside the mesh interpolate the separation surface.

This method introduces less nodes and tetrahedral elements than similar methods<sup>3;11</sup> and reduces the overhead on some key steps of the finite element method (e.g. solving the linear system at each time step), making it well-suited for real-time applications, such as augmented reality in surgery.

### 5.4 Handling internal structures

In our work we use the biomechanical model  $(P_V^t, C_V^t)$  of the virtual organ to update internal structures given by a pre-operative scan. For that, the initial positions  $P_I^0$  and the initial triangular surface connectivity  $C_I^0$  represent the internal structures like vessels, the urinary system, tumors or others. At initiation, barycentric coordinates are computed for each point  $p_{l,I}^0 \in P_I^0$  dependent on the shape functions used



**Fig. 3** Detection and simulation of cuts in silicone bands demonstrating the ability of our approach to distinguish between large strains (186% and 166%) and cuts: left: without augmentation, right: augmented with an uncut/cut model.

in the finite element formulation. Similar to the surface feature points, this helps to express the current position of the internal structures dependent on the biomechanical model, i.e.  $p_{l,I}^t = p_{l,I}^t(P_V^t) \in P_I^t(P_V^t)$ .

When the cut or the tearing of the organ yields a detection of a separation surface  $S$  as being described in subsection 5.2, we aim on propagating this change to the internal structures. In order to correctly update the internal structures, we update the connectivity  $C_I^t$  by deleting the triangles that intersect with the separation surface  $S$ , i.e.

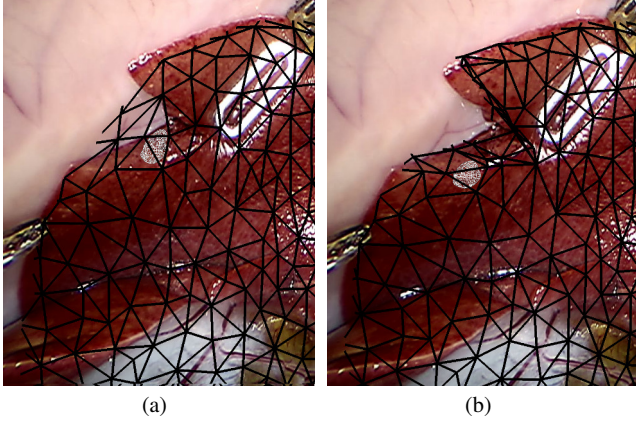
$$C_I^t = \{(c_{1,I}, c_{2,I}, c_{3,I}) | [p_{c_{1,I},I}^t, p_{c_{2,I},I}^t, p_{c_{3,I},I}^t] \cap S = \emptyset\} \quad (12)$$

## 6 Results

In this section, we demonstrate the potential of our approach to detect a (surgical) cut and to replicate the corresponding topological changes on a virtual model. Our experiments involve silicone data, *in vivo* liver data and *ex vivo* kidney data. All the results are obtained using a single view from a monocular camera. Qualitative and quantitative results are presented with *Dice's* coefficient for the first two datasets and more extensive comparisons are reported with the latter one.

Object	Dice's coefficient	
	Without cutting	With cutting
Silicone cut left	0.815	0.952
Silicone cut middle	0.900	0.964
Liver cut	0.906	0.963

**Table 1** Dice's coefficient on silicone data and *in vivo* liver. Our method results in higher Dice's coefficients when compared to methods not taking account of topological changes.



**Fig. 4** Augmented reality on cut and deformed liver overlaid by the virtual organ and a tumor; with a normal elastic model (a) and with our method (b)

### 6.1 Experiments on highly elastic silicone bands

Our algorithm was applied in two scenarios involving highly elastic silicone bands which are cut and then strongly deformed (see figure 3). The tracking of the features, the detection of the cut and the updates the deformable model due to topological changes are all performed in real-time on a single CPU computer.

### 6.2 Experiments on an *in vivo* liver

A second experiment was conducted from a video showing the cutting of a porcine liver lobe (see figure 4). A tumor has been inserted into the virtual representation of the organ. With no topology update, the augmented view of the tumor is distorted and misplaced, whereas with our cut detection and simulation, the tumor stays undeformed and correctly located below the cut.

### 6.3 Validation on an *ex vivo* kidney

The last experiment involves the cutting of an *ex vivo* kidney. It provides a challenging evaluation of our method, as the internal structures of the kidney (the calyces – a part of the

urinary system) have a complex geometry and cover a large part of the organ.

#### 6.3.1 Ground truth

In order to obtain a trustworthy ground truth for the kidney and its internal structures, two CT scans were performed before and after the manipulation. To ensure a good visualization of the internal structures in the CT images and to avoid a loss of volume when the cut occurs, we filled the calyces with a gel that solidifies. The gel contains  $BaSO_4$  microparticles, showing high contrast in CT images<sup>4</sup>.

The kidneys are cut perpendicularly to the long axis in the middle of the parenchyma, incising a part of the calyx system. The cut is widened by stretching the kidney along its longitudinal axis. The organs and internal structures have been segmented using active contour techniques (Snakes)<sup>10</sup>. The resulting surface meshes, displayed in the following, have been smoothed using nearest neighbor smoothing algorithm.

In order to quantify the obtained results, we measure the (sampled) Hausdorff distance  $H$  between the surface and the internal structures of the kidney comparing the simulated solution – with and without cutting – with the goal positions provided by the final CT scan. For the surface feature points we use the Euclidean norm:

$$\|p_{i,F}^t(P_V^t) - p_{i,F}^1\|_F = \|p_{i,F}^t(P_V^t) - p_{i,F}^1\|_2 \quad (13)$$

Averages and maximal values for the measures are collected in table 2.

#### 6.3.2 Our results

**Kidney 1** measures 102x27x55mm, the internal structures 58x11x27mm. We extracted 225 feature points, that are connected using a the radius  $r_{P_F^0} = 15$ mm. We applied our algorithm with a threshold  $\tau = 8$  to identify outliers when comparing to the average of the measure, inserting a new cut point  $p_C$  as soon as we have  $n = 8$  outliers and updating the neighborhood information using  $r_{p_C} = 7$ mm.

**Kidney 2** measures 96x27x56mm, the internal structures 52x12x32mm. We identified and tracked 219 feature points in the video stream and applied our method with  $r_{P_F^0} = 20$ mm,  $\tau = 10$ ,  $n = 3$ ,  $r_{p_C} = 13$ mm.

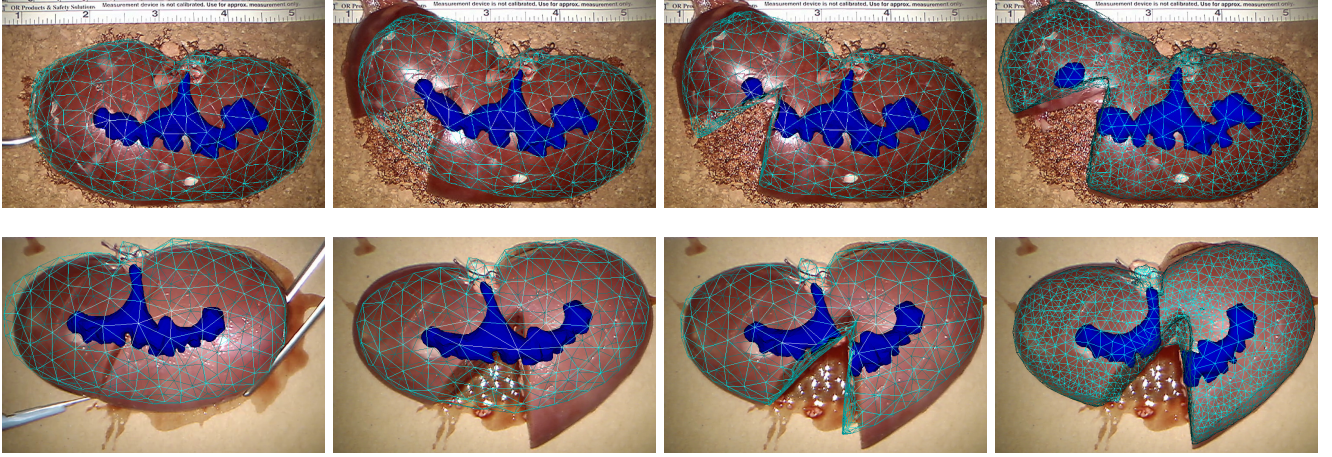
In both cases, the simulations were performed at minimally 25 fps, while on average the simulations run at 35 fps using a single CPU computer. The algorithms proposed in this work add less than 3% computational costs to a normal elastic FEM implementation, which is reflected in average by 0.15ms for the detection of a cut and 0.15-0.25ms for the update of the internal structures every time step.

For a visual comparison we refer to the overlaid images in figure 5. More details can be retrieved from figure 6,



Object	Entity	Measure	Average (mm)		Max (mm)	
			Without cutting	With cutting	Without cutting	With cutting
Kidney 1	$P_F^t(P_V^t)$	$\ \dots\ _F$	1.69	0.52	11.2	5.4
	$(P_{\partial V}^t(P_V^t), C_{\partial V}^t)$	$H$	1.76	1.49	13.1	10.1
	$(P_I^t(P_V^t), C_I^t)$	$H$	1.10	0.82	8.9	7.6
Kidney 2	$P_F^t(P_V^t)$	$\ \dots\ _F$	2.45	1.28	12.9	5.6
	$(P_{\partial V}^t(P_V^t), C_{\partial V}^t)$	$H$	1.96	1.22	13.1	6.9
	$(P_I^t(P_V^t), C_I^t)$	$H$	1.69	1.04	7.4	4.6

**Table 2** Evaluation on *ex vivo* kidney data. Our method results in a higher accuracy than standard approaches without cutting.



**Fig. 5** Augmented reality on cut and deformed kidney 1 (top) and 2 (bottom) overlaid by the virtual organ, the initial registration (left), final registrations: uncut (middle left), cut (middle right) and reference registration (right).

which displays the values of the sampled Hausdorff distance overlaid on the meshes. Table 2 summarizes the results by giving average and maximal values for the measures mentioned in section 6.3.1.

#### 6.4 Short discussion and additional information

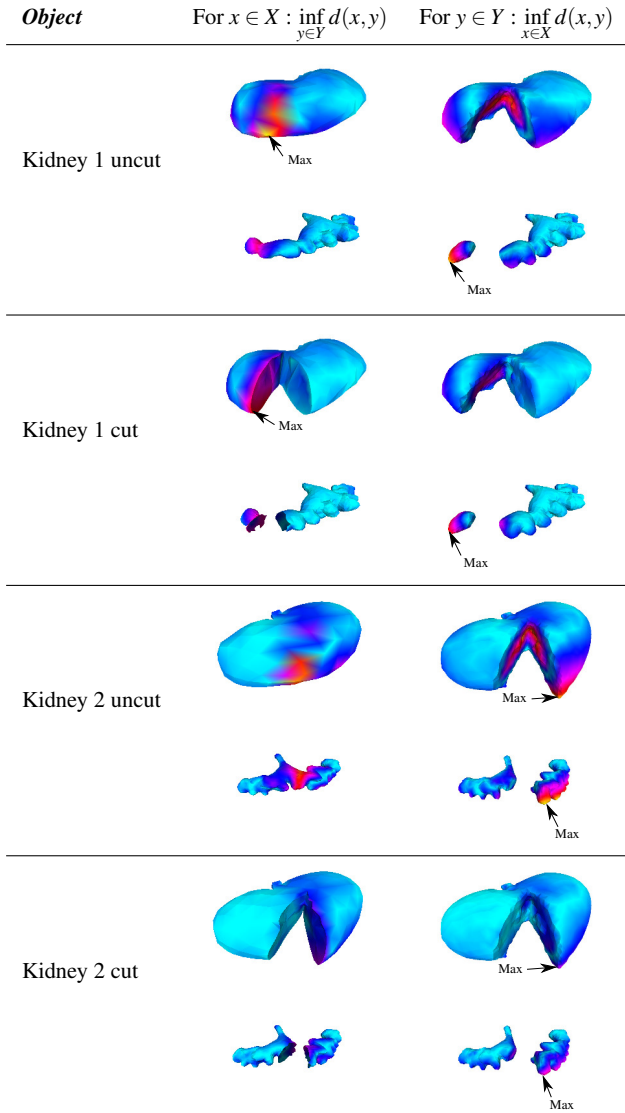
In the presented results, our method shows a clear advantage over existing approaches that do not account for (surgical) cuts. Particularly interesting is the impact of the parameters, that have to be adapted for the different scenarios – for example the radius  $r_{pc}$ : for a small  $r_{pc}$  (e.g. kidney 1), the cut advances in several steps (a progressing cut), while for a greater  $r_{pc}$  (e.g. kidney 2), cut lines are longer, more robust, but less precise.

The results are highly dependent on the deformation. Section 6.1 and previous work<sup>17</sup> shows the potentially positive impact of strongly deformed objects on the measure. Nevertheless, in the medical examples, we refrain from using strong deformations, to stay close to the medical workflow.

## 7 Limitations and discussion

**Clinical feasibility:** In the clinical context, the flow of features from a video stream is difficult to obtain, as surgical tools may block the direct view on the organ. On the other side, we do not benefit from information present in the images as the current state of the surgical intervention. For instance, combining our approach with the identification of surgical tools has a high potential and will be addressed in the future.

**Measure on the surface feature points:** Previous works<sup>17;18</sup> use similar measures. While<sup>17</sup> relies on the first component  $\delta_{lm,F}^t/\delta_{lm,F}^0$ , we use  $\delta_{lm,F}^1/\delta_{lm,F}^0$ . The choice of<sup>17</sup> yields a higher stability for the simulation as the underlying mechanical model regularizes or smooths strong movements or jumps of the surface feature points. Our measure is closer to what happens with the real object and is independent on the choice of the virtual model. Finally, the choice of the measure is dependent on the application and whether the surface feature points are trustworthy. Future measures will contain parts purely dependent on the virtual model, e.g. using the stress or the strain inside the biomechanical model, allowing for a better detection of a fracture.



**Fig. 6** Visualization of the two parts found in the Hausdorff distance, surfaces  $X$  of the simulated and  $Y$  the reference solution; low values in blue, high values in red/orange.

**Partial cuts:** For the separation surface  $S$  a fixed cut direction is given and does not change in the time of the simulation. Therefore the current version of our code cannot handle the detection of partial cuts in the direction of the camera. But a detection of a partial cut should be possible, when the cut volume is surrounded by surface feature points.

## 8 Conclusion

In this work we addressed the coupling of a pre-operative biomechanical model with the real organ, in such a way that topological changes such as surgical cuts, can be taken into account. As a result, the virtual model with internal struc-

tures like vessels, the urinary system or tumors – is kept coherent with the real organ even after an incision.

The virtual organ is deformed by taking account of its constitutive law and minimizing a stretching energy based on the tracking of surface feature points. From the motion of the different features, we retrieve information about the occurrence of a cut and the location where it has been performed. We then update the biomechanical model and its internal structures in real time.

Our evaluation shows the potential of our approach on examples ranging from *in vitro* over *in vivo* to *ex vivo*, using different measurements to compare our methods to the specific ground truths. Our experimental data is made available online in order to allow for a better comparison to future works<sup>2</sup>.

## 9 Acknowledgement and Statements

We thank Bruno Marques and Éric Leplat for assistance obtaining the experimental data, Etienne Schmitt for his ongoing support in the simulation of the topological changes and Igor Peterlik for the valuable comments improving this work.

The authors declare that they have no conflict of interest. All applicable international, national, and/or institutional guidelines for the care and use of animals were followed. This article does not contain any studies with human participants and thus does not contain patient data.

## References

1. Agudo, A., B. Calvo, and J. Montiel  
2012. Finite element based sequential bayesian non-rigid structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, Pp. 1418–1425. IEEE.
2. Bartoli, A. and A. Zisserman  
2004. Direct estimation of non-rigid registrations. In *British machine vision conference*, Pp. 899–908. BMVA.
3. Bielser, D., P. Glandon, M. Teschner, and M. Gross  
2004. A state machine for real-time cutting of tetrahedral meshes. *Graphical Models*, 66(6):398–417.
4. Diana, M., P. Halvax, D. Mertz, A. Legner, J.-M. Brulé, E. Robinet, D. Mutter, P. Pessaux, and J. Marescaux  
2015. Improving echo-guided procedures using an ultrasound-ct image fusion system. *Surgical innovation*, 22(3):217–222.
5. Feldmar, J., N. Ayache, and F. Betting  
1995. 3d-2d projective registration of free-form curves and surfaces. In *Computer Vision*, Pp. 549–556. IEEE.

<sup>2</sup> <http://www.open-cas.org/?q=PaulusIJCARS16>

6. Ferrant, M., A. Nabavi, B. Macq, P. M. Black, F. A. Jolesz, R. Kikinis, and S. K. Warfield  
2002. Serial registration of intraoperative mr images of the brain. *Medical image analysis*, 6(4):337–359.
7. Haouchine, N., J. Dequidt, M.-O. Berger, and S. Cotin  
2014. Single view augmentation of 3d elastic objects. In *Mixed and Augmented Reality (ISMAR)*, Pp. 229–236.
8. Haouchine, N., J. Dequidt, E. Kerrien, M.-O. Berger, and S. Cotin  
2012. Physics-based Augmented Reality for 3D Deformable Object. In *VRIPHYS - Virtual Reality Interaction and Physical Simulation*, Pp. 31–38, Darmstadt, Germany.
9. Haouchine, N., J. Dequidt, I. Peterlik, E. Kerrien, M.-O. Berger, and S. Cotin  
2013. Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, Pp. 199–208.
10. Kass, M., A. Witkin, and D. Terzopoulos  
1988. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331.
11. Koschier, D., S. Lipponer, and J. Bender  
2014. Adaptive tetrahedral meshes for brittle fracture simulation. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Pp. 57–66.
12. Leizea, I., H. Álvarez, I. Aguinaga, and D. Borro  
2014. Real-time deformation, registration and tracking of solids based on physical simulation. In *Mixed and Augmented Reality (ISMAR)*, Pp. 165–170.
13. Malti, A., R. Hartley, A. Bartoli, and J.-H. Kim  
2013. Monocular template-based 3d reconstruction of extensible surfaces with local linear elasticity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Pp. 1522–1529.
14. Moreno-Noguer, F., M. Salzmann, V. Lepetit, and P. Fua  
2009. Capturing 3d stretchable surfaces from single images in closed form. In *Computer Vision and Pattern Recognition CVPR*, Pp. 1842–1849.
15. Nicolau, S., L. Soler, D. Mutter, and J. Marescaux  
2011. Augmented reality in laparoscopic surgical oncology. *Surgical oncology*, 20(3):189–201.
16. Nienhuys, H.-W. and A. F. van der Stappen  
2001. A surgery simulation supporting cuts and finite element deformation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Pp. 145–152.
17. Paulus, C. J., N. Haouchine, D. Cazier, and S. Cotin  
2015a. Augmented reality during cutting and tearing of deformable objects. In *Mixed and Augmented Reality (ISMAR)*, Pp. 54–59.
18. Paulus, C. J., N. Haouchine, D. Cazier, and S. Cotin  
2015b. Surgical augmented reality with topological changes. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*.
19. Paulus, C. J., L. Untereiner, H. Courtecuisse, S. Cotin, and D. Cazier  
2015c. Virtual cutting of deformable objects based on efficient topological operations. *The Visual Computer*, 31(6-8):831–841.
20. Petit, A., V. Lippiello, and B. Siciliano  
2015a. Real-time tracking of 3d elastic objects with an rgb-d sensor. In *Intelligent Robots and Systems (IROS)*, Pp. 3914–3921.
21. Petit, A., V. Lippiello, and B. Siciliano  
2015b. Tracking fractures of deformable objects in real-time with an rgb-d sensor. In *International Conference on 3D Vision (3DV)*, Pp. 632–639.
22. Pilet, J., V. Lepetit, and P. Fua  
2008. Fast non-rigid surface detection, registration and realistic augmentation. *International Journal Computer Vision*, 76(2):109–122.
23. Pratt, P., D. Stoyanov, M. Visentini-Scarzanella, and G.-Z. Yang  
2010. Dynamic guidance for robotic surgery using image-constrained biomechanical models. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Pp. 77–85.
24. Salzmann, M. and P. Fua  
2011. Linear local models for monocular reconstruction of deformable surfaces. *Pattern Analysis and Machine Intelligence*, 33(5):931–944.
25. Salzmann, M., J. Pilet, S. Ilic, and P. Fua  
2007. Surface deformation models for nonrigid 3d shape recovery. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(8):1481–1487.
26. Sifakis, E. and J. Barbic  
2012. Fem simulation of 3d deformable solids: a practitioner’s guide to theory, discretization and model reduction. In *ACM SIGGRAPH 2012 Courses*, P. 20.
27. Suwelack, S.  
2013. *Real-Time Biomechanical Modeling for Intraoperative Soft Tissue Registration*. Karlsruhe Institute of Technology – Scientific Publishing.
28. Suwelack, S., S. Röhl, S. Bodenstedt, D. Reichard, R. Dillmann, T. dos Santos, L. Maier-Hein, M. Wagner, J. Wünscher, H. Kenngott, B. P. Müller, and S. Speidel  
2014. Physics-based shape matching for intraoperative image guidance. *Medical physics*, 41(11):111901.
29. Wu, J., R. Westermann, and C. Dick  
2014a. Physically-based simulation of cuts in deformable bodies: A survey. In *Eurographics State-of-the-Art Report*.
30. Wu, J., R. Westermann, and C. Dick  
2014b. Real-time haptic cutting of high resolution soft tissues. *Studies in Health Technology and Informatics (Proc. Medicine Meets Virtual Reality)*, 196:469–475.